REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
		FINAL - 01 APR 95 TO 31 MAR 96

4. TITLE AND SUBTITLE
COMPOSITIONAL MODELING FOR COMPUTER-BASED TUTORING OF PREDICTION TASKS

5. FUNDING NUMBERS
F49620-95-1-0295
61102F 2304/GS

6. AUTHOR(S)
BRUCE PORTER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
UNIVERSITY OF TEXAS
DEPARTMENT OF COMPUTER SCIENCE
AUSTIN, TEXAS 7812

AFOSR-TR-96
0445

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFOSR/NM
110 DUNCAN AVE, SUITE B115
BOLLING AFB DC 20332-8080

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE:
DISTRIBUTION UNLIMITED

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
SEE REPORT

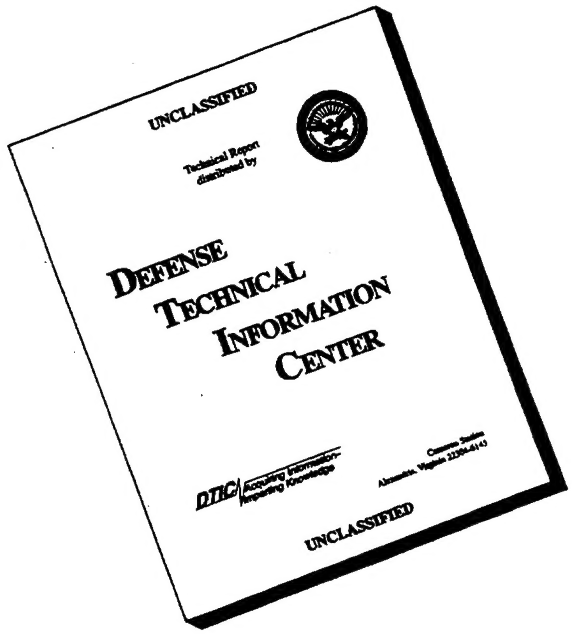
DTIC QUALITY INSPECTED 4

19961016 070

14. SUBJECT TERMS	15. NUMBER OF PAGES
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

Compositional Modeling for Computer-based Tutoring of Prediction Tasks

Final Technical Report[1]

Bruce Porter

Department of Computer Sciences
University of Texas at Austin
Austin, Texas 78712

Abstract

Our previous work has demonstrated the utility of computer-based advisory systems, such as expert systems and tutoring systems. (See our final report, dated 6/95, summarizing the results from our previous AFOSR contract.) We have developed methods for automatically answering a wide assortment of questions, even questions that were not anticipated when the advisory system was built. We have evaluated our question-answering methods by comparing their explanations with those written by human experts, using a "Turing test" experimental design. The results were very encouraging: a separate panel of human experts graded our machine-generated explanations only slightly lower than the human-generated ones.

Despite the effectiveness of computer-based advisory systems, a major obstacle prevents their widepread development and deployment: the knowledge base underlying each system is extremely difficult to build. Although we have built several knowledge bases in such diverse domains as legal reasoning and biology, each one was built "from scratch", with little transfer from other knowledge bases. Despite our considerable experience building such systems, our largest knowledge base required about ten man-years of sustained effort.

Our research during the past year has focused on this problem. We have developed methods for building knowledge bases from reusable components, analogous to the way that large "object-oriented" software systems are built. We are currently applying our methods to the task of building a knowledge base in the domain of Distributed Computing; furthermore, we plan to use these techniques for our next AFOSR project in the domain of hazardous-waste management. (See our research proposal, titled: "Improving Knowledge-Based Methods for Advisory and

1. Support for this research was provided by the Air Force Office of Scientific Research Contract # F59620-95-1-0295

1 Overview of the Research

A major cause of the "knowledge-engineering bottleneck" is that building one representation contributes little to building the next because each is idiosyncratic. We claim this problem is not inherent to knowledge engineering; rather, it is a limitation of current technology. Our research has developed a new suite of methods suitable for: specifying domain representations as compositions of abstract, reusable components; and assembling these representations, on demand, to answer questions.

Informally, a component is a description of an object, event, or state, represented as a system of concepts and relations that are packaged together and manipulated as a single unit. Although components might be expressed at any level of abstraction (from concrete instances to class prototypes), we focus on abstract ones, such as `container`, that can contribute to many domain representations. In the spirit of cliches, the `container` component describes an object that:

> partitions a space into two regions, `inside` and `outside`, permitting only two operations, `get` and `put`, to transport objects between these regions, such that the objects pass through the container's `portal`, subject to size and capacity constraints.

Although there are containers that violate this description (eg. sponges containing water, disks containing files), this does not reduce the need for reusable representations of typical containers, it only intensifies the need for methods that adapt representations.

The main challenge for a component-based representation falls on the composition operator: it must be capable of integrating information from various components, not simply collecting it. We implement this capability with *graph unification*, based on a similar algorithm for psi-term unification [?]. Our objective with unification is to syntactically merge graphs in a way that corresponds to merging semantically related information.

We are currently applying this representational framework in two ways. First, we are constructing an automated assistant for users of distributed computing systems, capable of answering novice users' questions.[2] The assistant contains three simple problem-solving algorithms. The first generates definitions of computing terminology by assembling component-based representations of domain concepts and converting them to text. The second performs diagnosis by constructing computing scripts and scanning them for failure points that explain the user's observations (iterating between data-gathering from the user and reasoning with information in the KB). The third generates short plans for achieving a user's goals (eg. "how do I reduce the minimum allowed password

[2]This project is partially funded by Digital Equipment Corporation, with a grant we obtained through the leverage of our AFOSR contract.

length") with a standard means-ends analysis algorithm, using planning operators built compositionally from the knowledge base. All three systems reason about a user's specific computing situation, represented as an instance graph in the knowledge base. This includes representation of the user's particular computing environment (eg. machines, their connectivity, processes running on them), and any particular activity which is being reasoned about. For example, if the user is asking about possible failures in a binding event (say), then an instance of binding event is created in the instance graph. The application systems query the knowledge base for particular pieces of information about the user's situation, and the query interpreter answers those queries by (lazily) unifying components with the instance graph.

Second, and more importantly, we are in the early stages of constructing a library of reusable components such as `communication`, `containment`, `exchange`, and `information`. While the library's contents are primarily being used as building blocks for the computing knowledge base (eg. a `database` is composed of `container`, `secure-item` and `resource`), our goal is to formalize the components in domain-general ways. Once completed, the library will facilitate building knowledge bases in a variety of domains.

2 Summary

A major obstacle to the widespread development and deployment of computer-based advisory systems is the "knowledge engineering bottleneck" — the difficulty of building the knowledge bases required by the advisory systems. To address this problem, we have developed a novel representation of components, evolved from a combination of conceptual graph theory and psi-term unification. We have illustrated how graph unification, used as a composition operator, can properly integrate components into a single structure. This approach offers a way to build domain-specific representations from reusable components.

Publications Resulting from this Research

1. J. Rickel and B. Porter, "Automated Modeling for Answering Prediction Questions: Selecting Relevant Influences", Fall Symposium on Relevance, American Association for Artificial Intelligence, 1994.

2. J. Rickel and B. Porter, "Automated Modeling for Answering Prediction Questions: Selecting the Time Scale and System Boundary", Proceedings of the National Conference on Artificial Intelligence, Seattle, Washington, 1994.

3. L. Acker and B. Porter, "Extracting Viewpoints from Knowledge Bases", Proceedings of the National Conference on Artificial Intelligence, Seattle, Washington, 1994.

4. J. Rickel and B. Porter, "Automated Modeling for Answering Prediction Questions: Selecting Relevant Influences", Fall Symposium on Relevance, American Association for Artificial Intelligence, 1994.

5. J. Rickel and B. Porter, "Automated Modeling for Answering Prediction Questions: Selecting the Time Scale and System Boundary", Proceedings of the National Conference on Artificial Intelligence, Seattle, Washington, 1994.

6. L. Acker and B. Porter, "Extracting Viewpoints from Knowledge Bases", Proceedings of the National Conference on Artificial Intelligence, Seattle, Washington, 1994.

7. J. Lester and B. Porter. "Interruption Handling:" Using Dynamic Replanning and Partially Refined Plans to Deal with Users' Interruptions. *AAAI Workshop on Planning for Interagent Communication*, pp. 71–77, AAAI-94, Seattle, Washington, 1994.

8. J. Lester and B. Porter. Designing Multi-Media Knowledge Delivery Systems: The Strong Representation Paradigm. *Symposium for Intelligent Multi-Media Multi-Modal Systems, AAAI Spring Symposium Series*, pp. 64–72, Stanford University, 1994.

9. B. Blumenthal and B. Porter, "Analysis and Empirical Studies of Derivational Analogy", Artificial Intelligence Journal, volume 67, number 2, pp. 287–328, June 1994.

10. C. Callaway and J. Lester. Robust Natural Language Generation from Large-Scale Knowledge Bases. *Proceedings of the Fourth Bar-Ilan Symposium on Foundations of Artificial Intelligence*, pp. 96–105, Jerusalem, Israel, 1995.

11. J. Lester and B. Porter. The KNIGHT Experiments: Empirically Evaluating an Explanation Generation System. *Symposium for Empirical Methods in Discourse Interpretation and Generation, AAAI Spring Symposium Series*, pp. 74–80, Stanford University, 1995.

12. J. Lester and B. Porter, "Developing and Empirically Evaluating Robust Explanation Generators: The KNIGHT Experiments", Computational Intelligence Journal, 1995, submitted.

13. P. Clark and B. Porter, "Building Domain Representations from Components", National Conference on Artificial Intelligence, 1996, submitted.

14. P. Clark and B. Porter, "A Compositional Approach to Representing Planning Operators", National Conference on Artificial Intelligence, 1996, submitted.

15. J. Rickel and B. Porter, "Automated Modeling of Complex Systems to Answer Prediction Questions", Artificial Intelligence Journal, 1996, submitted.